AMENDMENTS TO THE SPECIFICATION

Please replace the paragraph at page 15, line 17, with the following rewritten paragraph:

Also, the random number uniforming circuit 100 as shown in FIG. 1B is the same as the random number uniforming circuit 1 as shown in FIG. 1A, except that the number of bits to select the random number output from the shift register 200 is reduced to 6 bits, and an exclusive OR (XOR) circuit is added. That is, the random number uniforming circuit 1 as shown in FIG. 1B comprises the shift register 200 and the selector 300, in which the outputs of the exclusive OR circuit inputting the output of the selector 300 and binary random numbers ("0" or "1") are sequentially input into the data terminal D of the shift register 200, and shifted to the outputs Q00 to Q69 every time a reference pulse signal input into the clock terminal CLK of the shift register 200 rises. And the random numbers of 64 bits in the outputs Q00 to Q63 of the shift register 200 are input into the data terminals D00 to D63 of the selector 300, and the random numbers of 6 bits in the outputs Q64 to Q69 of the shift register ~~2~~ 200 are input into the addresses AD0 to AD5 of the selector 300. Thereafter, in the selector 300, one bit is selected from the random numbers of 64 bits input into the data terminals D00 to D63 in accordance with the address value of 6 bits input into the addresses AD0 to AD5 and output from the output terminal OUT.

Please replace the paragraph at page 20, line 5, with the following rewritten paragraph:

On the other hand, the random number uniforming circuit 1 as shown in FIG. 2A is the same as the random number uniforming circuit ~~1~~ 100 as shown in FIG. 1A, except that the number of bits to select the random number output from the shift register 200 is increased to 15 bits, and a combination of logical product (AND) circuits and exclusive OR (XOR) circuits is substituted for the selector 300. That is, the random number uniforming circuit ~~1~~ 100 as shown in FIG. 2A comprises the shift register 200, in which binary random numbers ("0" or "1") are sequentially input into the data terminal D of the shift register 200, and, shifted to the outputs Q00 to Q30 every time a reference pulse signal input into the clock terminal CLK of the shift register 200 rises. And the outputs of 15 logical product circuits inputting the random numbers of 15 bits in the outputs Q00 to

Q14 of the shift register 200 and the random numbers of 15 bits in the outputs Q16 to Q30 of the shift register 200 are sequentially synthesized with the output Q15 of the shift register 200 in the exclusive OR circuits and output.

Please replace the paragraph at page 21, line 11, with the following rewritten paragraph:

Also, the random number uniforming circuit 100 as shown in FIG. 2B is the same as the random number uniforming circuit ~~+~~ 100 as shown in FIG. 2A, except that the number of bits to select the random number output from the shift register 200 is reduced to 7 bits, and the exclusive OR (XOR) circuit is added. That is, the random number uniforming circuit ~~+~~ 100 as shown in FIG. 2B comprises the shift register 200, in which binary random numbers ("0" or "1") are sequentially input into the data terminal D of the shift register 200, and shifted to the outputs Q00 to Q14 every time a reference pulse signal input into the clock terminal CLK of the shift register 200 rises. And the outputs of seven logical product circuits inputting the random numbers of 7 bits in the outputs Q00 to Q06 of the shift register 200 and the random numbers of 7 bits in the outputs Q08 to Q14 of the shift register 200 are sequentially synthesized with the output Q07 of the shift register 200 in the exclusive OR circuits, finally synthesized with the original binary random number (raw data) in the exclusive OR circuit and output.

Please replace the paragraph at page 28, line 6, with the following rewritten paragraph:

Next, the testing by Poker Test is made. That is, the first counter 33 starts to count with the signal START_C generated via the control circuit 37 by the start signal (START) and the reference clock (CLK_0), and outputs the signal OUT_C at the time of 20,000 counts, as shown in FIG. 9. The shift register 34 converts serial random numbers (SRND) into the parallel random number ~~(PRND_4B)~~ (PRAND_4B) of 4 bits successively at the reference clock (CLK_0). The decoder 35 outputs the parallel random number to the output sections (SE_0 to SE_15) specified by the parallel random number ~~(PRND_4B)~~ (PRAND_4B), when the ENABLE signal generated via the control signal 37 by the start signal (START) and the reference clock (CLK_0) is active (once for every four

clocks). The counter 36 makes the initialization with the output signal CLR_CR of the control circuit 37, when the start signal (START) is entered, and when the ENABLE signal is active (once for every four clocks), the counter 36 specified by the decoder 35 counts up with the data of parallel random number ~~(PRND_4B)~~ (PRAND_4B). A sum of all the counters 36 amounts to 5,000 counts, and for serial random numbers generated in synchronism with the reference clock, the frequency distribution data (PokerData0 to PokerData15) for data (0 to 15) of the parallel random number ~~(PRND_4B)~~ (PRAND_4B) for every four bits is acquired at the time of 20,000 clocks after the start signal. The register 41 makes the initialization (POKD=0) with the output signal CLR_CR of the control circuit 37, when the start signal (START) is entered. After acquiring the frequency distribution data (PokerData0 to PokerData15), PokerData (POKD) is acquired by calculating a sum of squares of 16 frequency distribution data (PokerData0 to PokerData15) via the selector 38, the multiplier 39 and the adder 40. The comparator 42 compares the output PokerData (POKD) of the register 41 with the upper limit comparison data (e.g., 1,576, 928 bit) and the lower limit comparison data (e.g., 1,563,175 bit), and outputs the PokerJudge (POKJ) signal. Thereby, for serial random numbers generated in synchronism with the reference clock, PokerData and PokerJudge can be verified at the time of 20,000+16 clocks after the start signal.

Please replace the paragraph at page 31, line 5, with the following rewritten paragraph:

Finally, the testing by Long Runs Test is made. That is, the first counter 73 starts to count with the signal START_C generated via the control circuit ~~57~~ 76 by the start signal (START) and the reference clock (CLK_0), and outputs the signal OUT_C at the time of 20,000 counts, as shown in FIG. 12. The data holder 75 holds one bit of serial random number (SRND) successively at the reference clock (CLK_0). The comparator 74 compares the serial random number (SRND) with the random number held in the data holder 75, and outputs the signal CHANGE when the current random number is changed from the random number before one clock. The second counter 77 counts the clocks from the time when the signal CHANGE is output to the time when it is next output, and outputs the signal LRUNS_D. The second counter 77 makes the initialization (LRUNS_D=0) with the output signal CRL_CC of the control circuit 76, when the start signal

(START) is entered and when the signal CHANGE is output. The register 79 makes the initialization (LRUNS_D=0) with the output signal CLR_R of the control circuit 76, when the start signal (START) is entered. The first comparator 78 compares the output signal LongRunsData (LRND) of the register 79 with the output signal (LRUNS_D) of the second counter 77, and outputs the output signal COMP_U when LRND<LRUNS_D, outputs the LOAD_R signal via the control circuit 76 to the register 79, and successively holds the maximum value of LRUNS_D in the register 79. The second comparator 80 compares data with the upper limit comparison data (e.g., 26), and outputs a determination signal LongRunsJudge(LRNJ). The relationship between the signal LRUNS_D and the length (L) of the same signal as LRND is L=LRUNS_D+1, L(max)=LRND+1=LRUNS_D(max)+1. Thereby, for serial random numbers generated in synchronism with the reference clock, the data of LongRunsTest and the determination can be verified at the time of 20,000 clocks after the start signal.

Please replace the paragraph at page 40, line 8, with the following rewritten paragraph:

FIG. 19 is a circuit diagram of the physical random number generator according to another embodiment of the invention. This physical random number generator 101 comprises one integration circuit ~~104~~ 105 for integrating the clock signal through a resistor R and a capacitor C to output an integral waveform, two noise sources 106, two amplifiers 107 for amplifying the noise of the noise source 106 to output a noise signal, two mixers 108 for mixing the integral waveform and the noise signal, and two edge detection circuits 109 for detecting the first edge of jitter generated based on an output waveform of the mixer 108, as shown in FIG. 19. A D-type flip-flop 110 for outputting "0" or "1" based on a phase difference in the output signal between each edge detection circuit 109 is provided at the latter stage of each edge detection circuit 109. Furthermore, a D-type flip-flop 111 for synchronizing the random numbers with the clock signal is provided at the latter stage of the flip-flop 110.